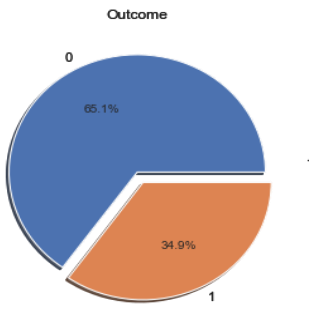
Classification Using Logistic Regression and Support Vector Machine

Analysis Report

INTRODUCTION

Diabetes is a chronic disease that affects a huge number of people and is accompanied by high blood sugar levels. Diabetes can be treated with lifestyle changes and/or minimal medication if identified early, therefore early detection is crucial. The goal of this report is to create a model that can produce more accurate predictions employing two of the most used classifiers in supervised machine learning techniques. Pregnancy, Glucose, Blood Pressure, Skin Thickness, Insulin, BMI, Diabetes Pedigree Function, and age are among the 8 attributes in the dataset. Furthermore, the 9th attribute (Outcome) is a class variable of each data point that indicates the diabetic outcome of '0' (negative) or '1' (positive). In the dataset, there are 268 diabetes individuals (34.9%) in the sample, and 500 non-diabetic patients (65.1%). The data imbalance is shown by the count plot graphical representation, which shows that the number of patients without diabetes is greater than those who do. In addition, at least one missing value (min. value indexed by 0) exists for each of the five attributes: insulin, glucose, BMI, skin thickness, and blood pressure. Furthermore, descriptive statistics show that the mean and median of blood pressure, body mass index, and skin thickness are nearly identical. The variable pedigree has the lowest standard deviation, whereas insulin has the greatest, meaning that pedigree has the least variability and insulin has the most.



Variable	Definition	Mean	Std. dev.	Median
Pregnancy	Frequency of pregnancy	3.85	3.37	3.00
Glucose	Concentration of plasma glucose (mg/dL)	121.66	30.44	117.00
BP	Diastolic blood pressure (mm Hg)	72.39	12.10	72.00
Skin	Tricep skinfold thickness (mm)	29.11	8.79	29.00
Insulin	Two-hour serum insulin (mu U/ml)	140.67	86.38	125.00
BMI	Body mass index (kg/m2)	32.46	6.88	32.30
Pedigree	A pedigree function for diabetes	0.47	0.33	0.37
Age	Age (log (years))	33.24	11.76	29.00

DATA PREPARATION

Exploratory Data Analysis (EDA), also known as Data Exploration, is a step in the Data Analysis process that uses a variety of approaches to better understand the dataset being used. In addition, a summary of the dataset (descriptive statistics) can be used to see how the dataset has been distributed for numerical values, such as the minimum, mean, percentiles, and maximum values. Furthermore, data must be cleaned, which includes removing duplicate values from the data frame, checking NULL values, and imputation of mean values for missing values. There are 5 attributes with a minimum value of 0, indicating that there are missing values. As a result, all zeros will be replaced with the attribute's mean values. The following model employs count plots, histograms, scatter plots, and pair plots to gain a better visual comprehension of the dataset. These

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
count	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000
mean	3.845052	120.894531	69.105469	20.536458	79.799479	31.992578	0.471876	33.240885	0.348958
std	3.369578	31.972618	19.355807	15.952218	115.244002	7.884160	0.331329	11.760232	0.476951
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.078000	21.000000	0.000000

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
count	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000
mean	3.845052	121.681605	72.254807	26.606479	118.660163	32.450805	0.471876	33.240885	0.348958
std	3.369578	30.436016	12.115932	9.631241	93.080358	6.875374	0.331329	11.760232	0.476951
min	0.000000	44.000000	24.000000	7.000000	14.000000	18.200000	0.078000	21.000000	0.000000

graphical annotations essentially provide specific dataset features such as dataset imbalance, skewedness of the set, and correlation matrices in order to comprehend the link between two variables. As a result, another strategy employed in this prediction model is Feature Scaling, which is used to standardize different features included in the data, usually to deal with variable magnitudes or values. In this case, the scaling technique has a significant impact on the accuracy of the classifier; without it, ML algorithms interpret bigger values to be higher and compact values to be lower, regardless of the unit of the values. Furthermore, libraries such as ScikitLearn, matplotlib, numpy, seaborn, pandas, and others simplify the creation of classification models by providing a large number of pre-defined functions. Splitting the data for testing and training is another crucial step in the data preparation phase. The complete data frame is separated into X and y in this paradigm, with X representing the data points in relation to the attributes and y representing the attributes. We have considered 80% of the data for training and 20% of the data for testing, so the split is of the following. Furthermore, experiments can be conducted by performing tweaks to the percentage of training data and testing data, to determine the impact this change on the accuracy of the classifier models.

```
df.shape
```

```
(768, 9)
```

```
X_train.shape, y_train.shape
```

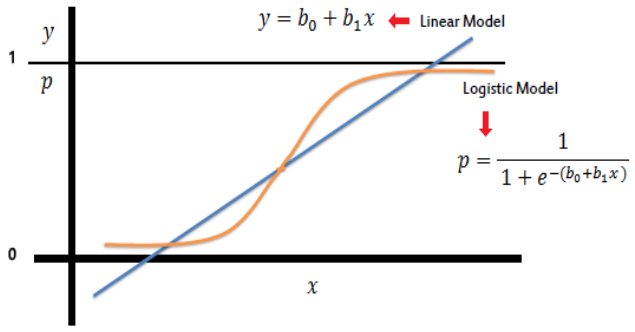
```
((614, 8), (614,))
```

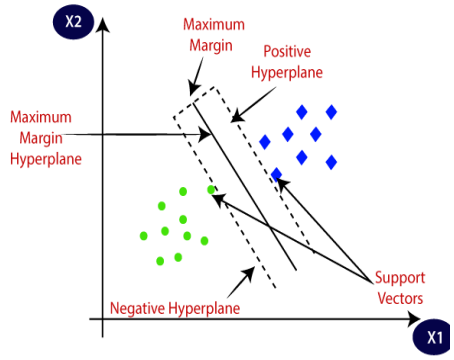
```
X_test.shape, y_test.shape
```

```
((154, 8), (154,))
```

CLASSIFIERS

Logistic Regression and Support Vector Machine are the two classifiers used in this model with the dataset provided. To develop an efficient and accurate model, first a thorough study and analysis of all the parameters of these two classifiers is required. Under the Supervised Learning approach, Logistic Regression is one of the most used Machine Learning algorithms. As a result, the conclusion must be categorical or discrete values. Primarily used for predicting the category department using a given collection of individual data points. The result can be expressed in terms of 0 or 1, yes or no, true or false, and so on. Except that Logistic Regression is generally employed to solve classification problems, it is nearly identical from Linear Regression. The most typical application of logistic regression is to estimate probabilistic values between 0 and 1. The model predicts that the instance belongs to the class if the estimated probability is more than 50%; otherwise, it predicts that it does not. Instead of constructing a regression line, in Logistic Regression, we fit an "S" shaped logistic function that predicts two maximum values: 0 and 1. This classifier's curve reflects the possibility of something happening, such as whether the cells are cancerous or not, if a mouse is obese or not based on its weight, and so on. In our scenario, whether or not someone has diabetes. The derived equation of the logistic model is → log [p/1-p] = B_o + B (Age). The S-form curve is





called the sigmoid function or the logistic function. The concept of a threshold value can be used to quantify the likelihood of 0 or 1. Support Vector Machine is a sophisticated and versatile Supervised Learning algorithm that may be used for classification and regression issues, as well as outlier detection. However, it is most typically employed for classification problems in machine learning. This classification model is particularly well adapted to classifying difficult little or medium-sized data frames. The purpose of this classifier algorithm is to find the best line or decision boundary that divides n-dimensional space into classes so that fresh data points can be readily placed in the appropriate category in the future. A hyperplane is the optimal choice boundary. The extreme points/vectors that assist create the hyperplane are chosen via SVM. Support vectors represent the extreme examples, which is why the method is called a Support Vector Machine. Take a look at the diagram, which shows two separate categories utilizing a choice or hyperplane. The margin is the distance between the hyperplane and the closest observations to the hyperplane (hard and soft). In this report, the results of Logistic Regression and SVM are expressed as a binary classification of '0' or '1'. A decision rule must be defined in order to classify a point as negative or positive. Both techniques are used in our model to assess classification and accuracy, using the data split percentages described before. The accuracies of the Logistic Regression and SVM models are 77.27 percent and 83.12 percent, respectively, in the case of an 80 percent training and 20 percent testing data split. Furthermore, as previously noted, changes to the percentages of the data frame split will affect the accuracy rates. Several observations have been made in this regard. As can be seen from the preceding facts, the test accuracy of the classifiers decreases as the amount of data available for training decreases. Overall, both categories clearly necessitate a sufficient amount of data for training in order to achieve efficient accuracy.

$$\vec{X}.\vec{w} - c \geq 0$$

$$\text{putting } -c \text{ as } b, \text{we get}$$

$$\vec{X}.\vec{w} + b \geq 0$$

$$\text{hence}$$

$$y = \begin{cases} +1 & \text{if } \vec{X}.\vec{w} + b \geq 0 \\ -1 & \text{if } \vec{X}.\vec{w} + b < 0 \end{cases}$$

	SCENARIOS	OBSERVATIONS (test accuracy)	
		LR	SVM
1.	Train data: 70% Test data: 30%	76.62%	79.65%
2.	Train data: 60% Test data: 40%	75.97%	77.27%

EVALUATION

In order to fully comprehend the findings of our test, anticipating the accuracy scores of the classifier models involved in the study and evaluating their performance is critical. This study covers and highlights some of the most widely utilised evaluation procedures in practise. Confusion-matrix, classification report, receiver operating characteristic (ROC), and area under the ROC curve analysis are the measures that are being focused on. A table of confusion metrics is used to describe how well a categorisation task performs. It compares projected and actual values to display the accuracy of a classifier. True Positive (TP), True Negative (TN), False Positive

PARAMETERS	LR	SVM
TN – True Negative	86	91
FP – False Positive	11	6
FN – False Negative	24	20
TP – True Positive	33	37

(FP), and False Negative (FN) are the four parameters that make up the Confusion Matrix. This is the most widely used metric for evaluating a model, however it is not a reliable indicator of its performance. It's especially unclear when there's a data imbalance. The following equation can be used to measure the accuracy of this evaluation model → Accuracy = (TP + TN) / (TP + FP + TN + FN). In our investigation, the LR and SVM accuracies are 0.77 and 0.83, respectively, but this is not a correct metric because false negative data points tamper with the actual accuracy. As a result, the dual concept of precision and recall might be applied at times. Additionally, the ScikitLearn package can be used to generate a classification report. It's a measure that shows the classification's precision, recall, F1 Score, and support, as well as macro and weighted average values. Finally, ROC and AUC scores can be calculated to provide a more accurate assessment of the model's performance. The ROC shows the TPR against the FPR at various threshold values, effectively separating the signal from the noise. The AUC is the measure of the ability of a classifier to distinguish between classes and is used as a summary of the ROC curve. The higher the AUC, the better the performance of the model at distinguishing between the positive and negative classes. In reference to code compiled with this report, the AUC for LR is 73.28 and for SVM, it is 79.36, indicating that the performance of the SVM model is better with a comparatively accurate distinction between positive and negative classes.

Classification Report of Logistic Regression:

	precision	recall	f1-score	support
0	0.7818	0.8866	0.8309	97
1	0.7500	0.5789	0.6535	57
accuracy			0.7727	154
macro avg	0.7659	0.7328	0.7422	154
weighted avg	0.7700	0.7727	0.7652	154

Classification Report of SVM:

	precision	recall	f1-score	support
0	0.8198	0.9381	0.8750	97
1	0.8605	0.6491	0.7400	57
accuracy			0.8312	154
macro avg	0.8401	0.7936	0.8075	154
weighted avg	0.8349	0.8312	0.8250	154

CONCLUSION

The major goal of this research was to design and implement Diabetes Prediction Using Machine Learning Approaches, as well as to analyze the performance of those methods, and it was effectively accomplished. In comparison, the SVM classifier has a greater accuracy of 83.12 percent, while the LR classifier has a lower accuracy of 77.27 percent. Regardless, even when the training and testing data ratios were changed, both models produced similar and competent accuracy scores. The reduced accuracy could be attributed to Logistic Regression's fragility and sensitivity to overfitting, which is considerably lower in the case of Support Vectors. Furthermore, we may deduce from the association matrix that people with a higher BMI, Skin Thickness, and Glucose levels are more likely to have diabetes. Patients with a high BMI and Skin Thickness, with a correlation of more than 0.5, are considered to be more prone to or likely to develop diabetes. Support Vector Machines have shown to be one of the most efficient algorithms for creating prediction models. This research also reveals that, in addition to method selection, additional aspects like as data pretreatment, the elimination of redundant and null values, normalization, cross-validation, feature selection, and the use of ensemble approaches can increase model accuracy and runtimes.